班　级：　机研 2302
学　号：　2023200724

北京化工大学

深度学习课程报告

题 目：基于 RBF-PCA 的故障诊断系统

学院名称：机电工程学院

专　　业：动力工程及工程热物理

学生姓名：尹胜基

2024 年 1 月 5 日

摘 要

随着“工业 4.0”和“智能制造 2025”的推广，智能装备正成为高端装备未来的发展方向。在这个过程中，故障诊断识别系统作为一项重要的研究领域，可以实时抑制机械的不平衡振动，从而实现装备的自主健康。这种系统不仅可以提高设备的运行效率和生产能力，还能够降低设备维修的成本和风险。因此，研究和开发这种系统对于推动智能制造和高端装备制造的发展具有重要意义。

大型旋转机械的状态监测与故障诊断是现代企业生产管理的必备手段，通过科学的监测和诊断，可以实现设备的高效、安全、可靠运行，为企业的可持续发展提供有力保障。本课题以旋转机械转子系统典型振动故障为研究主体，考虑对其进行分析，建立旋转机械转子系统典型故障诊断智能识别系统。本文主要研究内容与结论如下：

（1）本文通过查阅相关文献对当前的机械智能故障诊断算法和运用神经网络进行故障分类的研究方法进行了总结探究，并最终决定选择 RBF 神经网络进行故障诊断分类；

（2）使用 PCA 降维技术对提取特征后的数据进行降维处理，来降低数据的维度来解决“维数灾难”问题，使数据更加可靠；

（3）通过使用人工神经网络中的径向基神经网络建立故障分类软件，利用其对故障通过分类器识别并进行相关实验验证，完成分类软件设计；

（4）搭建试验台采集数据进行相关实验验证，实验表明，诊断准确率达到 85%以上。

关键词：典型振动故障；PCA 降维；RBF

第 1 章 绪论

通过状态监测与故障诊断，可以及时发现设备的异常运行情况，避免设备因故障造成停机维修，从而减少生产损失和维修成本以及提高生产效率。在大型旋转机械的状态监测与故障诊断中，应用先进的传感器、信号处理和数据分析技术，可以实现对设备的高效、精准监测和诊断。同时，还可以通过云计算、物联网等手段，提高设备的运行效率。总之，大型旋转机械的状态监测与故障诊断是现代企业生产管理的必备手段，通过科学的监测和诊断，可以实现设备的高效、安全、可靠运行，为企业的可持续发展提供有力保障。

现代工业生产自动化程度高，机械设备规模、结构和工作强度逐渐增大，长时间运转可能面临复杂多样的故障。因此，对旋转机械故障诊断提出更高的要求。

就旋转机械方面的故障诊断而言，国际上已经有了相关的研究成果。例如，美国的 Sohre J · S · 发表论文总结了旋转机械的典型故障特点和原因，并被广泛应用于高速旋转机械故障诊断领域。在国内方面，高金吉博士深入探索了高速旋转机械的故障机制，并基于此，建立起一种以 10 类 58 种为基础的故障类型的分类模型，以此来更好地掌握其发生的可能性。

关于旋转设备故障诊断的方法, 主要包括如下几种：常规诊断法、信息融合方法、采用专家系统、采用神经网络、采用模糊方法和可能性理论的方法等。这些方法可以帮助我们更准确地诊断机械故障，提高故障处理效率和准确性。

转子不平衡是旋转机械振动的一个重要影响因素，特别是对于高速旋转机械。同时，转子不平衡也会影响其它转子故障的发生和加剧。为了解决这一问题，现有在线自动平衡系统可以实时抑制机床不平衡振动，采用电磁式自动平衡方式，能够在短时间内抑制机床主轴的不平衡振动。这种自动平衡系统可以通过测量机床的振动信号，计算出机床的不平衡情况，并通过控制电磁力的大小和方向来实现对机床的自动平衡。通过引入先进的自动平衡技术，我们能够大幅改善机器的运行状态，并且能够更准确地控制零件的位置。这种技术不仅能够更好地保证设备的正常运行，还能够大幅提升生产效率，带来更多的经济收入。

本课题拟对旋转机械转子系统典型故障振动系统进行开发，通过研究降维方法的学习使用，神经网络的构建与验证提出旋转机械转子系统故障诊断的方案，以便提高旋转机械的工作效率，保障其安全可靠运行。

第 2 章 RBF 和 PCA 理论基础

2.1 PCA 降维

首先介绍下为什么要进行数据降维：

通过对高维数据的分解，可以使其变得更加简单、清晰。这种方法的优点包括：（1）节约运算成本费用：分解高维数据所消耗的存储空间和运算资源较小，从而节约了运算成本费用；（2）改善模型性能：分解高维数据使得模型变得简单，从而避免了过拟合的问题；（3）去除冗余信息：通过采取数据降维技术，我们能够消除多维度的数据，从而改善模型的精度与可靠性；（4）可视化展示：高维数据难以进行可视化展示，而低维数据可以更容易地进行可视化展示，有助于人们更好地理解和分析数据；（5）数据压缩：通过采取数据压缩技术，将多维度的数据维度降低，从而节约了存储空间。

PCA（Principal Component Analysis，主成分分析）作为一种高级的降维算法，它可以有效的把复杂的高维数据分解成更加精细的维度，从而更加精准的抓取出数据的关键特征。PCA 算法旨在通过线性变换，把原有的数据投影至一个全局的位置，以达到减少其变形量的目的，并且可以有效地提高投影的精度，以达到有效的数据降维。PCA 算法的优点是可以消除数据中的冗余信息和噪声，提高模型的准确性和解释性。同时，PCA 算法也可以用于数据压缩和可视化等方面的应用。

PCA 降维的基本原理如下图 2-1 所示：

首先对原始数据进行去均值化操作，使得数据中心为原点。经过计算，首先可以得出协方差矩阵，它描述了各个维度之间的相互关系。通过将协方差矩阵的特征值进行分解，就可以更加清晰地了解每一个特征值和它们的相关向量。为了更好地表示原始数据，可以考虑将其投影到一个新的矩阵上，从而实现降维处理。在 PCA 算法中，特征向量是 PCA 算法的核心，它代表了数据在新坐标系下的方向。特征向量的长度代表了数据的方差，方向则代表了数据的主要变化方向。通过 PCA 算法，就可以实现从原始数据中选择前k个最大的特征值，并将其转换成具有较高方差的特征向量，从而有效地实现数据的降维。这样，最后就可以将具有较高方差的特征维度保留下来，而将具有较低方差的特征维度排除在外，从而达到降维的目的。

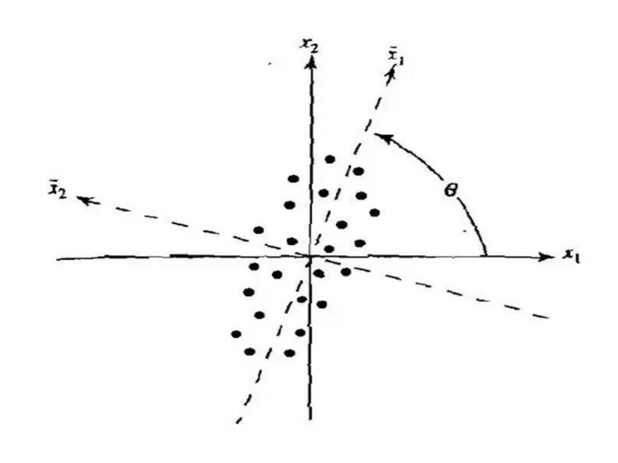


图 2-1 PCA 降维原理图

实现 PCA 的一种算法如下：

输入:数据集 $X=\{x_1,x_2,x_3,\ldots,x_n\}$，需要降到$k$维。

（1）去平均值，即所有特征减去 $\bar{x}$ 。

（2）计算协方差矩阵$\frac{1}{n}XX^T$，这里处理样本数量n或$n-1$与否,与特征向量无关。

（3）解决协方差矩阵$\frac{1}{n}XX^T$的关键问题，包括确定其特征值和方向。

（4）特征值排序，将对应的按大小顺序排列的前 k 个特征向量按行排列为特征向量矩阵 P。

（5）通过将k条特征值映射至一组新的空间，使得 Y=PX 成为可能。

实现步骤画出流程图如下：

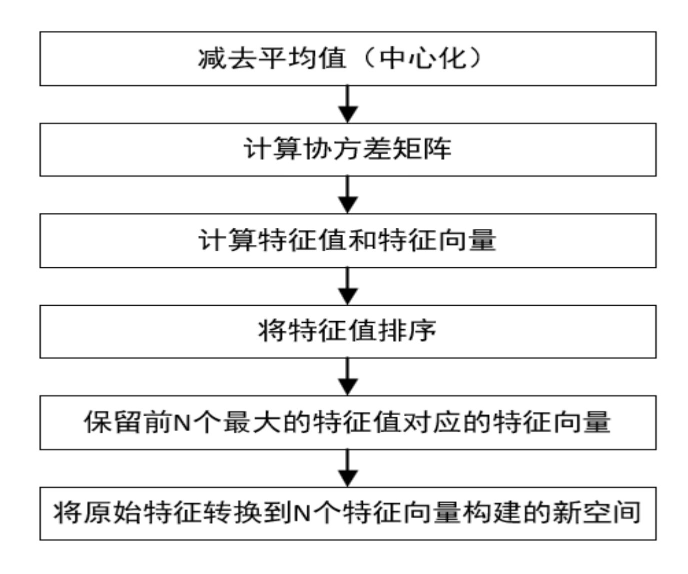


图 2-2 PCA 降维实现算法

其中，样本均值：

$$\overline{x}=\frac{1}{n}\sum_{i=1}^{N}x_i$$

样本方差：

$$S^2=\frac{1}{n-1}\sum_{i=1}^{n}\left(x_i-\overline{x}\right)^2$$

样本 X 和样本 Y 的协方差：

$$\begin{aligned}\operatorname{cov}(X,Y) &= E[(X-E(X))(Y-E(Y))] \\ &= \frac{1}{n-1}\sum_{i=1}^{n}\left(x_i-\overline{x}\right)\left(y_i-\overline{y}\right)\end{aligned}$$

2.2 径向基神经网络（RBF）

RBF 神经网络的架构包含三个基本单元：输入、处理器、以及反馈。在处理器的基本单元中，输入是指从系统获取的数据，它会被转换成一系列的模型，然后被处理器处理。在处理器的基本单元（Radial Basis Function）中，它们会使用径向基函数来处理高维数据。径向基函数是一类常见的数学方法，其中包含高斯函数和多项式函数。反馈层的节点数通常与目标变量的维度相同，用于输出网络的预测结果。反馈层的节点通常采用线性函数作为激活函数，将处理层的输出进行线性组合，得到最终的预测结果。RBF 神经网络的显著特性在于它的模型准确性和高效性，它的模型能够很好地模拟任何连续函数，并且在短时间内就能获得准确的结果，避免了模型的局限性。另外，它还适合在复杂的情况下进行模型的训练；不仅如此，对于多分类问题，可以使用一次训练多个输出。

RBF 神经网络结构如图 2-3 所示。

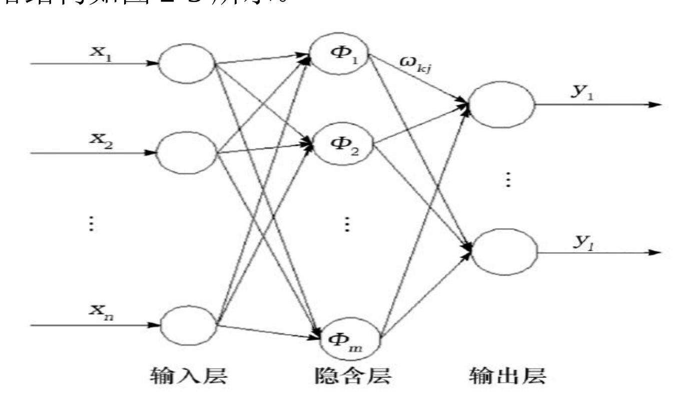


图 2-3 RBF 神经网络结构

当输入向量 $\boldsymbol{X}=[x_1,x_2,\cdots,x_n]$，输出向量 $\boldsymbol{Y}=[y_1,y_2,\cdots,y_l]$ 时，RBF 神经网络的隐含层和结果为：

$$y_j=\sum_{k=1}^{m}\omega_{kj}\Phi_k(x) \quad j\in 1,2,\cdots,l$$

$$\Phi_k(x)=\exp\left(-\frac{\|\boldsymbol{X}-\boldsymbol{c}_k\|^2}{2\sigma^2}\right)$$

其中: σ 为基函数宽度；ω_{kj} 为隐含层第 k 个神经元与输出层第 j 个神经元之间的连接权值；$\boldsymbol{c}_k$ 为第 k 个神经元的中心点向量；$\Phi_k(x)$ 为第 h 个径向基函数的 Gauss 函数。

第 3 章 基于 PCA-RBF 的代码实现与实验验证

3.1 基于 MATLAB 的 PCA 特征降维

在 MATLAB 中可以通过编写上述算法代码来实现 PCA 数据降维，如下图 3-1 所示：

```
function S=princa(X)
[m,n]=size(X); %计算矩阵的行m和列n
 %-------------第一步：标准化矩阵-----------------%
mv=mean(X); %计算各变量的均值
st=std(X); %计算各变量的标准差
X=X-repmat(mv,m,1); %标准化矩阵X
 %-------------第二步：计算相关系数矩阵-----------------%
% R1=X'*X/(m-1); %方法一：协方差矩阵计算公式
% R2=cov(X);      %方法二：协方差矩阵计算函数
R=corrcoef(X); %方法三：相关系数矩阵函数
%-------------第三步：计算特征向量和特征值-----------------%
[V,D]=eig(R);        %计算矩阵R的特征向量矩阵V和特征值矩阵D,特征值由小到大
      %将特征向量矩阵V从大到小排序
  %将特征值矩阵由大到小排序
E=diag(D);           %将特征值矩阵转换为特征值向量
%-------------第四步：计算贡献率和累计贡献率-----------------%
ratio=0; %累计贡献率
for k=1:n
    r=E(k)/sum(E);   %第k主成份贡献率
    ratio=ratio+r;  %累计贡献率
    if(ratio>=0.8)  %取累计贡献率大于等于90%的主成分
        q=k;
        break;
    end
end
%-------------第五步：计算得分-----------------%
V=V(:,1:q);
S=X*V;
```

图 3-1 PCA 降维算法 MATLAB 实现

在 MATLAB 中也可直接调用 *pca* 函数实现 PCA 数据降维，其调用格式如下：

$$[coeff, score, latent] = pca(X)$$

两者降维效果无明显差异，本课题选用调用 *pca* 函数实现 PCA 数据降维。

3.2 基于 MATLAB 的径向基神经网络故障分类实例

在 MATLAB 中画出 RBF 径向基传递函数如下图 3-2 所示：

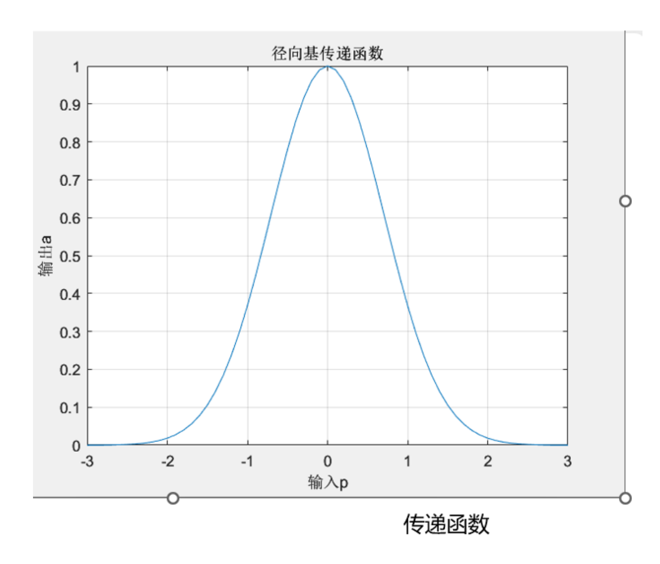


图 3-2 径向基传递函数（Gauss 函数）

RBF 神经网络的训练过程主要包括以下几个步骤：①确定隐含层节点数：通过交叉验证等方法，确定隐含层节点数，以提高神经网络的性能和泛化能力。②初始化权重：随机初始化隐含层和输出层的权重，以便开始训练神经网络。③计算径向基函数：通过非线性变换处理输入数据，计算径向基函数的值，以获得更准确的结果。④计算输出层权重：通过最小二乘法等方法，计算输出层的权重，以产生最终的输出结果。⑤更新权重：通过反向传播算法等方法，更新神经网络中的权重，以提高神经网络的性能和准确性。

在 MATLAB 中可调用函数 *newrb* 实现 RBF，其调用格式如下：

$$net = newrb(P, T, goal, spread, MN, DF)$$

式中部分参数解释如下表所示：

表 3-1 RBF 参数解释

参数	实际意义
P	输入向量
T	目标分类向量
goal	均方误差目标，默认为 0
spread	径向基函数的扩展速度，默认为 1
MN	径向基神经元最大值
DF	径向基增加步长，默认为 25
net	返回值，即一个 RBF 网络

通过构建 3 维数据集，我们从 300 组不同的不平衡和不对中的情况下收集各 300 组信息，以及 300 组的干扰，构建出 $P = 900\times 3$ 维的模型，以期望能够更好地模拟故障诊断情况。将 P 输入集随机分成 540 组和 360 组，分别用于进行训练和评估。

RBF 神经网络分类部分使用 MATLAB 编写的部分代码如下：

```
% 2. 随机产生训练集和测试集
temp = randperm(size(feature,1));
% 训练集——900*0.6个样本
P_train = feature(temp(1:540),:)';        %P和T列代表个数必须保持相等
T_train = labal(temp(1:540),:)';
% 测试集—360个样本
P_test = feature(temp(541:end),:)';
T_test = labal(temp(541:end),:)';
N = size(P_test,2);

%% III. RBF神经网络创建及仿真测试
%%
% 1. 创建网络
t = cputime;
rbf_net=newrb(P_train,T_train,0.04,0.02,150,1);
%%
% 2. 仿真测试
T_sim = sim(rbf_net,P_test);
result=round(T_sim);
 t = cputime - t;
%% IV. 性能评价
%%
% 1. 相对误差error
%计数判断正确则为1，否则为0
no_test=size(result,2);
count=0;
for n=1:1:no_test
if result(:,n)==T_test(:,n)
   count=count+1;
else

end
end
accuracy=count/no_test;
```

图 3-3 RBF 神经网络分类 MATLAB 实现

在 MATLAB 中对其进行训练，如下图 3-4 所示：

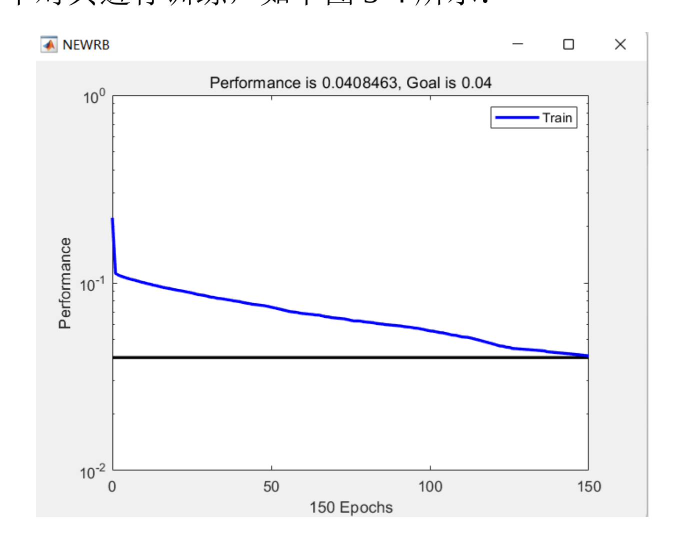
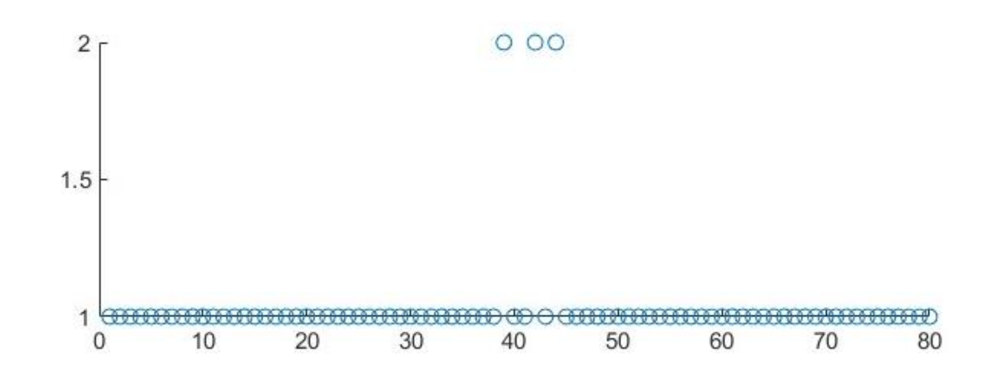


图 3-4 RBFNN 的训练过程图

3.3 实验验证

把采集的不平衡状态、不对中状态下数据结合之前编的算法进行算法验证。

采用上述典型的振动智能诊断算法，可以有效地检测出不平衡和不对中的数据，划分训练集测试集之后，从中抽取各 80 组数据，不对中故障标签设置为 1，不平衡故障标签设置为 2，并将其结果展示在图 3-5 中，从而更加清晰地了解它们的状态，如下图所示：



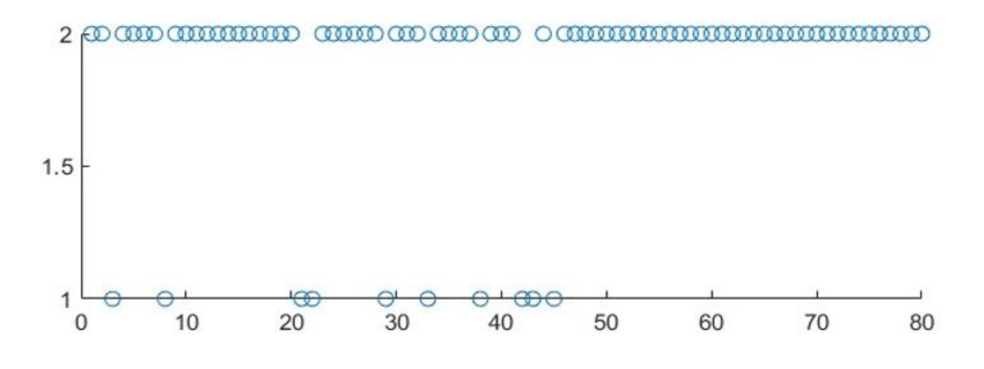


图 3-5 不对中（上）和不平衡（下）故障诊断识别结果

经过多次实验验证分析，可以发现典型的振动智能诊断软件能够准确地识别出不对中和不平衡的故障情况，其平均识别率达到 85%以上，完全符合课题的预期要求。

第 4 章 结论

本文以旋转机械转子系统典型振动故障为研究主体，从理论、仿真和试验等多个方面建立旋转机械转子系统典型故障诊断智能识别系统并进行相关实验验证，并取得了良好的实验效果。

对提取出的特征进行 PCA 降维处理等操作，并在 MATLAB 软件中编写相关代码实现这一功能；在完成特征降维之后，需要选择合适的分类器进行故障分类。本文通过对人工神经网络中的多种分类器网络进行网络结构探究、隐含层传递函数比较等进行多种人工神经网络的优缺点对比分析，并最终选择径向基神经网络（RBF）作为该课题的故障分类器。之后对降维后的数据进行神经网络的训练与验证，并在 MATLAB 中编写相关代码完成这一功能。

最后，为验证算法的效率和准确性以及集成方案的可行性，考虑做相关实验进行实验验证。将不平衡、不对中这两种情况下的故障数据进行采集，然后，带入到原来的故障诊断算法中进行验证，故障诊断率达到 85%以上，符合预期要求。